

Docket No.: R2184.0255/P255  
(PATENT)

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re Patent Application of:  
Hiroki Kobayashi

Application No.: 10/647,503

Confirmation No.: 2701

Filed: August 26, 2003

Art Unit: N/A

For: IMAGE PROCESSING APPARATUS  
HAVING WEB SERVER FUNCTION

Examiner: Not Yet Assigned

**CLAIM FOR PRIORITY AND SUBMISSION OF DOCUMENTS**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Dear Sir:

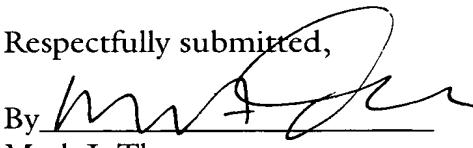
Applicant hereby claims priority under 35 U.S.C. 119 based on the following prior foreign applications filed in the following foreign countries on the dates indicated:

Country	Application No.	Date
Japan	2002-258190	September 3, 2002
Japan	2003-292492	August 12, 2003

In support of this claim, a certified copy of each said original foreign application is filed herewith.

Dated: December 15, 2003

Respectfully submitted,

By   
Mark J. Thronson

Registration No.: 33,082  
DICKSTEIN SHAPIRO MORIN &  
OSHINSKY LLP  
2101 L Street NW  
Washington, DC 20037-1526  
(202) 785-9700  
Attorney for Applicant



Japan Patent Office

This is to certify that the annexed is a true copy of the following application as filed with this Office.

Date of Application: September 3, 2002

Application Number: Japanese Patent Application  
No.2002-258190

[ST.10/C]: [JP2002-258190]

Applicant(s): RICOH COMPANY, LTD.

August 4, 2003

Commissioner,  
Japan Patent Office

Yasuo Imai (Seal)

Certificate No.2003-3062169

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2002年 9月 3日  
Date of Application:

出願番号 特願2002-258190  
Application Number:

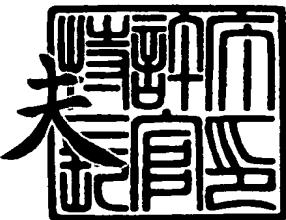
[ST. 10/C] : [JP2002-258190]

出願人 株式会社リコー  
Applicant(s):

2003年 8月 4日

特許庁長官  
Commissioner,  
Japan Patent Office

今井康夫



【書類名】 特許願  
【整理番号】 0205456  
【提出日】 平成14年 9月 3日  
【あて先】 特許庁長官 太田 信一郎 殿  
【国際特許分類】 G03G 21/00  
【発明の名称】 画像処理装置  
【請求項の数】 26  
【発明者】  
【住所又は居所】 東京都大田区中馬込1丁目3番6号 株式会社リコー内  
【氏名】 小林 寛樹  
【特許出願人】  
【識別番号】 000006747  
【氏名又は名称】 株式会社リコー  
【代理人】  
【識別番号】 100070150  
【弁理士】  
【氏名又は名称】 伊東 忠彦  
【手数料の表示】  
【予納台帳番号】 002989  
【納付金額】 21,000円  
【提出物件の目録】  
【物件名】 明細書 1  
【物件名】 図面 1  
【物件名】 要約書 1  
【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 画像処理装置

【特許請求の範囲】

【請求項 1】 ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、

前記書式情報データを格納する第二の記憶手段と、前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、

該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することを特徴とする画像処理装置。

【請求項 2】 前記複数の書式情報ファイルはXMLであることを特徴とする請求項1記載の画像処理装置。

【請求項 3】 前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するまでに、前記書式情報ファイルを解凍することを特徴とする請求項1又は2記載の画像処理装置。

【請求項 4】 前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信した際に、前記書式情報ファイルを解凍することを特徴とする請求項1又は2記載の画像処理装置。

【請求項 5】 前記解凍手段は、前記第一の記憶手段が格納する前記書式情報ファイルのうち、前記圧縮されている複数の書式情報ファイルのうち全てを解凍することを特徴とする請求項3又は4記載の画像処理装置。

【請求項 6】 前記解凍手段は、前記端末からWebページの送信要求を受信する際に、要求された前記Webページを生成するために使用される前記書式情報ファイルを解凍することを特徴とする請求項1乃至4いずれか一項記載の画像処理装置。

**【請求項 7】** 前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用された時期が近いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする請求項6記載の画像処理装置。

**【請求項 8】** 前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用頻度が高いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする請求項6記載の画像処理装置。

**【請求項 9】** 前記Webページ生成手段が前記Webページを生成した後、前記Webページの生成に使用された前記書式情報データは、前記第二の記憶手段から削除されることを特徴とする請求項6記載の画像処理装置。

**【請求項 10】** 前記Webページ生成手段は、要求された前記Webページの生成に使用される前記書式情報データが既に前記第二の記憶手段に格納されているかを判断する解凍済み判断手段を有し、

前記Webページ生成手段は、前記解凍済み判断手段の判断結果に基づいて前記書式情報データを使用して前記Webページを生成することを特徴とする請求項6乃至8いずれか一項記載の画像処理装置。

**【請求項 11】** 前記Webページ生成手段は、前記書式情報データの数が所定の値を超えた場合は、いずれかの前記書式情報データを前記第二の記憶手段から削除することを特徴とする請求項10記載の画像処理装置。

**【請求項 12】** 前記Webページ生成手段は、使用されていない期間が最も長い前記書式情報データを削除することを特徴とする請求項11記載の画像処理装置。

**【請求項 13】** 前記Webページ生成手段は、解凍された時期が最も早い前記書式情報データを削除することを特徴とする請求項11記載の画像処理装置。

**【請求項 14】** ネットワークを介して接続される端末からの要求を受信し

、前記端末に表示される情報を表すWebページを前記端末に送信する画像処理装置によるWebページ生成方法であって、

圧縮されている書式情報ファイルを書式情報データに解凍する書式情報解凍手順と、

前記書式情報データを使用してWebページを生成するWebページ生成手順とを有することを特徴とするWebページ生成方法。

**【請求項15】** 前記解凍手順は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するまでに実行されることを特徴とする請求項14記載のWebページ生成方法。

**【請求項16】** 前記解凍手順は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信した際に、実行されることを特徴とする請求項14記載のWebページ生成方法。

**【請求項17】** 前記解凍手順は、圧縮されている全ての前記書式情報ファイルを解凍することを特徴とする請求項15又は16記載のWebページ生成方法。

**【請求項18】** 前記解凍手順は、前記端末からWebページの送信要求を受信する際に、要求された前記Webページの生成に必要な前記書式情報ファイルを解凍することを特徴とする請求項14乃至16いずれか一項記載のWebページ生成方法。

**【請求項19】** 前記Webページ生成手順において、前記Webページを生成した後、

前記Webページの生成に使用された前記書式情報データは削除することを特徴とする請求項18記載の画像処理装置。

**【請求項20】** 要求された前記Webページの生成に使用される前記書式情報ファイルが既に前記書式情報データに解凍されているかを判断する解凍済み判断手順を有し、

前記Webページ生成手順は、前記解凍済み判断手順の判断結果に基づいて前記書式情報データを使用して前記Webページを生成することを特徴とする請求項18記載のWebページ生成方法。

【請求項21】 画像処理装置と端末とがネットワークを介して接続された画像処理システムであって、

前記画像処理装置は、前記端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、

該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、

前記書式情報データを格納する第二の記憶手段と、

前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、

該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信し、

前記端末は、前記Webページを表示する表示手段を有することを特徴とする画像処理システム。

【請求項22】 前記複数の書式情報ファイルはXMLであることを特徴とする請求項21記載の画像処理装置システム。

【請求項23】 ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、

該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、

前記書式情報データを格納する第二の記憶手段と、

前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、

該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することを特徴とする情報処理装置。

【請求項24】 前記複数の書式情報ファイルはXMLであることを特徴とする請求項23記載の情報処理装置。

【請求項 25】 情報処理装置と端末とがネットワークを介して接続された情報処理システムであって、

前記情報処理装置は、前記端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、

圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、

該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、

前記書式情報データを格納する第二の記憶手段と、

前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、

該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信し、

前記端末は、前記Webページを表示する表示手段を有することを特徴とする情報処理システム。

【請求項 26】 前記複数の書式情報ファイルはXMLであることを特徴とする請求項 25 記載の情報処理システム。

#### 【発明の詳細な説明】

##### 【0001】

##### 【発明の属する技術分野】

本発明は、画像処理装置に関し、特にWebサーバの機能を有する画像処理装置に関する。

##### 【0002】

##### 【従来の技術】

近年では、インターネット等のネットワークを介して交換する文書データは、その文書構造と書式情報を分離させて扱うことが一般的になりつつある。文書データの文書構造と書式情報を分離することで、コンピュータによる文書データの情報処理が容易となり、また一つの文書に様々な書式情報を与えてレイアウトし、再利用することができる。

##### 【0003】

文書データを文書構造と書式情報へ分離した例としてXML（eXtensible Markup Language）とXSL（eXtensible Stylesheet Language）とがある。一般的なWebサーバにおいては、インターネット上に公開するWebページに掲載する情報としてXMLデータを生成あるいは保持する。Webサーバは、Webクライアントからの要求を受けて、XSLデータを用いてXSLT（XSL Transformations）変換を行うことでHTML（HyperText Markup Language）データを生成し、Webクライアントに送信する。

#### 【0004】

一方、最近では画像処理装置においてもWebサーバとしての機能を有するものが存在し、例えば画像処理装置のステータスや、保管された文書情報、スケールされているジョブ、FAXやe-mail等に使用されるアドレス帳を汎用的なWebブラウザから見ることができる。

#### 【0005】

##### 【発明が解決しようとする課題】

しかしながら、XSLファイルはファイルサイズが大きくなりがちであり、画像処理装置のようにメモリ資源（ROM）が限られている機器において種々のXSLファイルを保持しておくのは非常に困難であるという問題がある。

#### 【0006】

即ち、一般的にXSLファイルのようなテキストファイルには、改行、タブ及びインデント等のように可読性を向上させるための制御コードが入力されることが多いため、ファイルサイズが大きくなってしまう。

#### 【0007】

XSLファイルによるメモリ資源の圧迫は、大容量のハードディスクを備えることにより解決することも可能であるが、その場合には、画像処理装置のコストアップにつながるという問題がある。

#### 【0008】

本発明は、上記の点に鑑みてなされたものであって、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる画像処理装置の提供を目的とする。

**【0009】****【課題を解決するための手段】**

そこで上記課題を解決するため、本発明は、請求項1に記載されるように、ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、前記書式情報データを格納する第二の記憶手段と、前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することを特徴とする。

**【0010】**

このような画像処理装置では、Webページを生成するための書式情報を圧縮して記憶手段に格納しておくため、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる。

**【0011】**

また、本発明は、請求項2に記載されるように、前記複数の書式情報ファイルはXMLであることを特徴とする。

**【0012】**

このような画像処理装置では、書式情報の定義として広く利用されているXML (eXtensible Stylesheet Language) を利用するため、Webページの生成処理の開発を効率的に行うことができる。

**【0013】**

また、本発明は、請求項3に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するまでに、前記書式情報ファイルを解凍することを特徴とする。

**【0014】**

このような画像処理装置では、端末から最初のWebページの送信要求を受信するより前に予め書式情報ファイルを解凍しておくため、端末の要求に対するレ

レスポンスの悪化を防止することができる。

#### 【0015】

また、本発明は、請求項4に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信した際に、前記書式情報ファイルを解凍することを特徴とする。

#### 【0016】

このような画像処理装置では、端末から最初のWebページの送信要求を受信した際に書式情報ファイルを解凍するため、それまで解凍先の記憶領域の容量を節約することができると共に、以降の端末の要求に対するレスポンスの悪化を防止することができる。

#### 【0017】

また、本発明は、請求項5に記載されるように、前記解凍手段は、前記第一の記憶手段が格納する前記書式情報ファイルのうち、前記圧縮されている複数の書式情報ファイルのうち全てを解凍することを特徴とする。

#### 【0018】

このような画像処理装置では、予め全ての書式情報ファイルを解凍しておく。従って、以降の端末からのWebページの送信要求に対応したWebページの生成に必要な書式情報ファイルは必ず解凍されており、端末の要求に対するレスポンスの悪化を防止することができる。

#### 【0019】

また、本発明は、請求項6に記載されるように、前記解凍手段は、前記端末からWebページの送信要求を受信する際に、要求された前記Webページを生成するために使用される前記書式情報ファイルを解凍することを特徴とする。

#### 【0020】

このような画像処理装置では、端末からWebページの送信要求を受信する際に、要求されたWebページの生成に必要な書式情報ファイルを解凍するため、解凍先の記憶領域の容量を節約することができる。

#### 【0021】

また、本発明は、請求項7に記載されるように、前記解凍手段は、前記画像処

理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用された時期が近いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする。

#### 【0022】

このような画像処理装置では、使用された時期が近いものから順に所定の数の書式情報ファイルを解凍しておくため、解凍先の記憶領域の容量を節約することができると共に、最近要求されたWebページの送信要求に対してレスポンスの悪化を防止することができる。

#### 【0023】

また、本発明は、請求項8に記載されるように、前記解凍手段は、前記画像処理装置の起動後であって前記端末から最初の前記Webページの送信要求を受信するより前又は受信した際に、前記第一の記憶手段が格納する前記書式情報ファイルのうち、使用頻度が高いものから順に所定の数の前記書式情報ファイルを解凍することを特徴とする。

#### 【0024】

このような画像処理装置では、使用頻度が高いものから順に所定の数の書式情報ファイルを解凍しておくため、解凍先の記憶領域の容量を節約することができると共に、要求される頻度の高いWebページの送信要求に対してレスポンスの悪化を防止することができる。

#### 【0025】

また、本発明は、請求項9に記載されるように、前記Webページ生成手段が前記Webページを生成した後、前記Webページの生成に使用された前記書式情報データは、前記第二の記憶手段から削除されることを特徴とする。

#### 【0026】

このような画像処理装置では、Webページの生成に使用した書式情報データは、解凍先の記憶領域から削除されるため、解凍先の記憶領域の容量を節約することができる。

#### 【0027】

また、本発明は、請求項10に記載されるように、前記Webページ生成手段は、要求された前記Webページの生成に使用される前記書式情報データが既に前記第二の記憶手段に格納されているかを判断する解凍済み判断手段を有し、前記Webページ生成手段は、前記解凍済み判断手段の判断結果に基づいて前記書式情報データを使用して前記Webページを生成することを特徴とする。

#### 【0028】

このような画像処理装置では、要求されたWebページの生成に必要な書式情報ファイルが既に解凍されている場合は、解凍された書式情報データを使用してWebページを生成するため、端末に対するレスポンスの悪化を防止することができる。

#### 【0029】

また、本発明は、請求項11に記載されるように、前記Webページ生成手段は、前記書式情報データの数が所定の値を超えた場合は、いずれかの前記書式情報データを前記第二の記憶手段から削除することを特徴とする。

#### 【0030】

このような画像処理装置では、書式情報データの数が所定の値を超えた場合は、いずれかの書式情報データを解凍先の記憶領域から削除するため、解凍先の記憶領域の容量を節約することができる。

#### 【0031】

また、本発明は、請求項12に記載されるように、前記Webページ生成手段は、使用されていない期間が最も長い前記書式情報データを削除することを特徴とする。

#### 【0032】

このような画像処理装置では、使用されていない期間が最も長い書式情報データを削除するため、最近解凍した書式情報データを利用したWebページの要求に対するレスポンスの悪化を防止することができる。

#### 【0033】

また、本発明は、請求項13に記載されるように、前記Webページ生成手段は、解凍された時期が最も早い前記書式情報データを削除することを特徴とする

。

### 【0034】

このような画像処理装置では、解凍された時期が最も早い書式情報データを削除するため、最近解凍した書式情報データを利用したWebページの要求に対するレスポンスの悪化を防止することができる。

### 【0035】

更に、上記課題を解決するための手段として、本発明は、上記画像処理装置によって処理されるWebページ生成方法及び上記画像処理装置によって構成される画像処理システムとすることもできる。

### 【0036】

#### 【発明の実施の形態】

以下、図面に基づいて本発明の実施の形態を説明する。

### 【0037】

図1は、本発明の実施の形態における画像処理装置の機能構成例を示す図である。図1において、画像処理装置10は、スキャナ1231、プリンタ1232及びFAX1233等の個別の画像処理を行うためのハードウェア資源と、エンジン制御ボード1201と、エンジンI/F1202と、OS1203と、API(Application Program Interface)1204と、スキャナアプリ1211、プリンタアプリ1212、ファックスアプリ1213及びWebアプリ1214等の各種アプリケーションを有している。

### 【0038】

エンジン制御ボード1201は、スキャナ1231、プリンタ1232、FAX1233等の画像処理装置10の個別機能を制御する。エンジンI/F1202は、エンジン制御ボード1201に対するインターフェースを、OS1203やAPI1204に提供する。

### 【0039】

OS1203は、UNIX(登録商標)などのオペレーティング・システムであり、スキャナアプリ1211、プリンタアプリ1212等、アプリケーションとして実装されている各種ソフトウェアをプロセスとして並列実行する。

### 【0040】

A P I 1 2 0 4 は、スキャナアプリ1211、プリンタアプリ1212等のアプリケーションへのインターフェースを提供する。

### 【0041】

スキャナアプリ1211は、スキャナ用アプリケーションであり、プリンタアプリ1212は、ページ記述言語（P D L）、P C L 及びポストスクリプト（P S）を有するプリンタ用アプリケーションである。ファックスアプリ1213は、ファクシミリ用アプリケーションである。また、W e b アプリ1214は、画像処理装置10にネットワークを介して接続されるW e b クライアントに対して画像処理装置10の機能を提供するためのアプリケーションである。

### 【0042】

このように、O S 1 2 0 3 及びA P I 1 2 0 4 上に各種アプリケーションを搭載するよう構成しているので、各画像処理の処理部分を共有することが可能となる。従って、画像処理装置10全体の生産性を高めるとともに、各アプリケーションの開発効率を高めることができる。

### 【0043】

次に本発明の第一の実施の形態について説明する。図2は、本発明の第一の実施の形態におけるW e b サーバとしての画像処理装置の機能構成例を示す図である。図2において画像処理装置10は、R O M 1 1 、R A M 1 2 、初期化プログラム13、解凍プログラム14、W e b サーバプログラム15、W e b アプリケーション16、及びX S L T プロセッサ17等から構成される。

### 【0044】

R O M 1 1 は、圧縮されたX S L (eXtensible Stylesheet Language) ファイル等を格納し、R A M 1 2 は、解凍プログラム14により解凍されたX S L データ等を格納する。

### 【0045】

初期化プログラム13は、画像処理装置10が起動する際に呼び出され、解凍プログラム14を呼び出す。

### 【0046】

解凍プログラム14は、ROM11に格納されている圧縮されたXSLファイル（以下、「圧縮XSLファイル」という。）をXSLデータとしてRAM12上に解凍する。なお、解凍プログラム14は、必ずしも画像処理装置10の起動と同時に初期化プログラムから呼び出される必要はなく、Webクライアント100からWebサーバプログラム15に最初のWebページの送信要求（HTTP（HyperText Transfer Protocol）リクエスト）があるまでに呼び出されればよい。従って、例えば後述するWebサーバプログラム15がその起動時に解凍プログラム14を呼び出してもよい。

#### 【0047】

Webサーバプログラム15は、一般にHTTPd（HTTP daemon）と呼ばれるプログラムに相当するものであり、画像処理装置10をWebサーバとして機能させるためのものである。Webサーバプログラム15は、Webクライアント100からのHTTPリクエストに応じて、Webアプリケーション16を起動する。またWebサーバプログラム15は、Webアプリケーション16が生成したHTML（HyperText Markup Language）データを要求元のWebクライアント100に送信する。なお、Webサーバプログラム15は、図1において、API1204の一部の機能として画像処理装置10に実装されている。

#### 【0048】

Webアプリケーション16は、Webクライアント100に対し、Webページを通して種々のサービスを提供するプログラムである。例えばWebアプリケーション16は、画像処理装置10が管理している文書情報の提供や印刷処理等のサービスを提供する。Webアプリケーション16は、Webサーバプログラム15から呼び出され、サービスの処理結果をXML（eXtensible Markup Language）データとして生成し、更にXSLTプロセッサ17によりXMLデータをHTMLデータ（Webページ）に変換し、Webサーバプログラム15に対して出力する。なお、Webアプリケーション16は、図1において、Webアプリ1214に相当するものである。

#### 【0049】

XSLTプロセッサ17は、Webアプリケーション16から呼び出され、そ

の際に指定されるXMLデータを、RAM上12に格納されているXSLデータを利用して、HTMLデータに変換する。

#### 【0050】

なお、Webクライアント100は、画像処理装置10とネットワークを介して接続される端末であり、Webブラウザを備えている。

#### 【0051】

以下、図2の画像処理装置10の処理手順について説明する。図3は、第一の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

#### 【0052】

ステップS10において、画像処理装置10は起動されると、初期化プログラム13を呼び出す。

#### 【0053】

ステップS10に続いてステップS11に進み、初期化プログラム13は解凍プログラム14を呼び出す。ステップS11に続いてステップS12に進み、解凍プログラム14は、解凍すべき全ての圧縮XSLファイルファイル名を確認する。圧縮XSLファイルのファイル名は、例えば所定のファイルに全ての圧縮XSLファイルのファイル名の一覧を管理しておけばよい。その場合、解凍プログラム14はそのファイル（以下「圧縮XSL一覧ファイル」という。）を参照することで全ての圧縮XSLファイルのファイル名を確認することができる。

#### 【0054】

ステップS12に続いてステップS13に進み、解凍プログラム14は、解凍対象とする圧縮XSLファイルのうちの一つをROM11から読み込み、更に解凍後のXSLデータを格納するための領域をRAM12に確保する（S14）。

#### 【0055】

ステップS14に続いてステップS15に進み、解凍プログラム14は圧縮XSLファイルを解凍し、XSLデータとしてRAM12に格納する。ステップS15に続いてステップS16に進み、解凍プログラム14は、解凍対象とする圧縮XSLファイルの全てを解凍したかどうかを判断する。まだ解凍すべき圧縮X

SLファイルがROM11にある場合はステップS13に進み、解凍プログラム14は解凍処理を継続する。既に解凍対象とする圧縮XSLファイルの全てを解凍した場合は、解凍処理は終了する。

#### 【0056】

この後、画像処理装置100がWebクライアント100からのHTTPリクエストを受信すると、XSLTプロセッサ17は、RAM12に格納されたXSLデータを利用して、Webアプリケーション16が生成したXMLデータの変換処理を実行する。

#### 【0057】

上述したように第一の実施の形態における画像処理装置10によれば、ROM11には圧縮されたXSLファイルを格納するため、ROM11の容量を節約することができる。また、Webクライアント100からの最初のHTTPリクエストが送信される前に圧縮XSLファイルを解凍しておくため、Webクライアント100に対するレスポンスの悪化を防止することができる。

#### 【0058】

なお、RAM12の容量が少ない等の制約がある場合は、圧縮XSL一覧ファイルに全ての圧縮XSLファイルのファイル名を登録しておくのではなく、所定のものに限定して登録しておけばよい。この場合、画像処理装置10がWebクライアント100からのHTTPリクエストを受信した時点で、要求されるWebページに対応したXSLデータがRAM12内に存在しない状況が発生するが、その場合は要求されるWebページに対する圧縮XSLファイルをその場で解凍すればよい。要求されたWebページに対応する圧縮XSLファイルの解凍については第三の実施の形態において説明する。また、要求されたWebページに対応する圧縮XSLファイルが既に解凍済みかどうかの判断の方法については第四の実施の形態において説明する。

#### 【0059】

なお、使用頻度の高いものから上位幾つか、又は使用された時期が近いものから上位幾つか等を解凍しておけば、要求されるWebページに対応したXSLデータがRAM12内に存在しない状況の発生の頻度を低く抑えることができる。

使用頻度については、圧縮XSL一覧ファイルに各圧縮XSLファイルの使用数のカウンタ（以下「使用数カウンタ」という。）を設け、XSLTプロセッサ17がXSLデータを使用するたびに、そのXSLデータの使用数カウンタをインクリメントすればよい。また、使用された時期については、同じく圧縮XSL一覧ファイルに各圧縮XSLファイルの使用時期を記録する項目を設け、XSLTプロセッサ17がXSLデータを使用するたびに、その使用時期を更新すればよい。解凍プログラム14は、使用数カウンタ又は記録された使用時期を参照することで、使用頻度の高い圧縮XSLファイル、又は使用された時期が近い圧縮XSLファイルを特定することができる。

#### 【0060】

次に本発明の第二の実施の形態について説明する。図4は、本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第二の実施の形態における画像処理装置20は、第一の実施の形態における画像処理装置10とほぼ同様の構成である。即ち、ROM21はROM11に、RAM22はRAM12に、解凍プログラム24は解凍プログラム14に、Webサーバプログラム25はWebサーバプログラム15に、Webアプリケーション26はWebアプリケーション16に、XSLTプロセッサ27はXSLTプロセッサ17にそれぞれ対応している。但し、画像処理装置20は、初期化プログラム11に相当するものは有しておらず、Webサーバプログラム25が解凍プログラム24の呼び出しを行う。

#### 【0061】

以下、図4の画像処理装置20の処理手順について説明する。図5は、第二の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

#### 【0062】

ステップS20において、画像処理装置20のWebサーバプログラム25は、Webクライアント100からHTTPリクエストを受信すると、解凍済みフラグが1であるかどうかを確認することにより、全ての圧縮XSLファイルが解凍済みであるかどうかを判断する（S21）。

**【0063】**

解凍済みフラグが1の場合、即ち既に全ての圧縮XSLファイルを解凍済みの場合はステップS28に進む。解凍済みフラグが1でない場合、即ち圧縮XSLファイルの解凍処理を実行していない場合は、ステップS22に進む。

**【0064】**

ステップS22において、Webサーバプログラム25は解凍プログラム24を呼び出す。ステップS23からステップS27において、解凍プログラム24は第一の実施の形態におけるステップS12からステップS16における手順と同様の手順によってROM21内の全ての圧縮XSLファイルを解凍し、RAM22に格納する。全ての圧縮XSLファイル21の解凍が完了するとステップS28に進む。ステップS28においてWebサーバプログラム25は、解凍済みフラグの値を1にし、ステップS29に進む。

**【0065】**

ステップS29において、Webサーバプログラム25がWebクライアント100からのHTTPリクエストに対応したWebアプリケーション26を呼び出すと、Webアプリケーション26は所定の処理を実行し、その処理結果としてXMLデータを生成する。ステップS29に続いてステップS30に進み、Webアプリケーション26は、XSLTプロセッサ27を呼び出し、処理結果としてのXMLデータをHTMLデータのWebページに変換し、HTMLデータをWebサーバプログラム25に対して出力する。ここでXSLTプロセッサ27は、RAM22上に格納されたXSLデータを利用してXMLデータをHTMLデータに変換する。

**【0066】**

ステップS30に続いてステップS31に進み、Webサーバプログラム25は、HTMLデータとして生成されたWebページをWebクライアント100に送信する。

**【0067】**

上述したように、第二の実施の形態における画像処理装置20によれば、ROM21には圧縮されたXSLファイルを格納するため、ROM21の容量を節約

することができる。また、画像処理装置20は、Webクライアント100からのHTTPリクエストの受信時であって解凍済みフラグが1でない場合にのみ解凍処理を実行する。ここで、解凍済みフラグは一度解凍処理が実行されると1に設定されるものである。即ち、画像処理装置20は、起動後にWebクライアント100からのHTTPリクエストを最初に受信した際に圧縮XSLファイルを解凍する。従って、画像処理装置20は、Webクライアント10からの最初のHTTPリクエストを受信するまでRAM22の容量を節約することができると共に、以降のWebクライアント100からのHTTPリクエストに対するレスポンスの悪化を防止することができる。

#### 【0068】

なお、解凍対象を所定のものに限定してもよく、その実現方法については、第一の実施例の場合と同様である。

#### 【0069】

次に本発明の第三の実施の形態について説明する。図6は、本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第三の実施の形態における画像処理装置30は、第二の実施の形態における画像処理装置20と同様の構成である。即ち、ROM31はROM21に、RAM32はRAM22に、解凍プログラム34は解凍プログラム24に、Webサーバプログラム35はWebサーバプログラム25に、Webアプリケーション36はWebアプリケーション26に、XSLTプロセッサ37はXSLTプロセッサ27にそれぞれ対応している。但し、画像処理装置30において、解凍プログラム34を呼び出すのは、XSLTプロセッサ37である。

#### 【0070】

以下、図6の画像処理装置30の処理手順について説明する。図7は、第三の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

#### 【0071】

ステップS40において、画像処理装置30のWebサーバプログラム35は、Webクライアント100からHTTPリクエストを受信する。ステップS4

0に続いてステップS41に進み、Webサーバプログラム25がHTTPリクエストに対応するWebアプリケーション36を呼び出すと、Webアプリケーション36は所定の処理を実行し、処理結果としてXMLデータを生成する。

#### 【0072】

ステップS41に続いてステップS42に進み、Webアプリケーション26は、処理結果としてのXMLデータをHTMLデータに変換すべく、XMLデータとそれに対応するXSLファイルを指定してXSLTプロセッサ37を呼び出す。ここで、使用するXSLファイルについては、Webアプリケーション36のロジックとして組み込んでおいてもよい（ハードコーディング）が、Webアプリケーション36が図8に示すXSLファイル対応テーブルを参照することにより判断するようにしてもよい。

#### 【0073】

図8は、圧縮XSLファイル対応テーブルの一例を示す図である。XSL対応テーブル361は、Webクライアント100から要求されたURLである「URL」と、それに対して利用する圧縮XSLファイル名である「XSL」とをデータ項目として有している。Webアプリケーション36は、対応テーブル361を参照することにより、例えばtopPage.cgiのURLが指定された場合は、topPage.xls.zを使用すべきことを判断し、そのファイル名を指定してXSLTプロセッサ37を呼び出す。

#### 【0074】

ステップS42に続いてステップS43に進み、XSLTプロセッサ37は、XSLT変換に必要なXSLデータ入手すべく、Webアプリケーション36から指定された圧縮XSLファイルを指定して、解凍プログラム34を呼び出す。

#### 【0075】

ステップS44及びステップS45において、解凍プログラム34は、第二の実施の形態におけるステップS24及びステップS25における手順と同様の手順で、ROM31内にある指定された圧縮XSLファイルを解凍し、RAM32に格納する。

**【0076】**

ステップS45に続いてステップS46に進み、XSLTプロセッサ37は、RAM32に格納されたXSLデータを使用してXMLデータをHTMLデータに変換し、Webページを生成する。ステップS46に続いてステップS47に進み、XSLTプロセッサ37は、RAM32において解凍されたXSLデータが格納されている領域を解放することにより、XMLデータをRAM32内から削除する。

**【0077】**

ステップS47に続いてステップS48に進み、Webサーバプログラム35は、Webクライアント100に対し生成されたWebページを送信する。

**【0078】**

上述したように、第三の実施の形態における画像処理装置30によれば、ROM31には圧縮されたXSLファイルを格納するため、ROM31の容量を節約することができる。また、画像処理装置30は、Webクライアント100からのHTTPリクエストを受信した際に、HTTPリクエストの応答としてのWebページの生成に必要な圧縮XSLファイルのみをRAM32に解凍し、Webページの生成が完了したらRAM32におけるXSLデータの領域を解放するため、RAM32の領域を節約することができる。

**【0079】**

次に本発明の第二の実施の形態について説明する。図9は、本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。第四の実施の形態における画像処理装置40は、第三の実施の形態における画像処理装置30と同様の構成である。即ち、ROM41はROM31に、RAM42はRAM32に、解凍プログラム44は解凍プログラム34に、Webサーバプログラム45はWebサーバプログラム35に、Webアプリケーション46はWebアプリケーション36に、XSLTプロセッサ47はXSLTプロセッサ37にそれぞれ対応している。

**【0080】**

以下、図9の画像処理装置40の処理手順について説明する。図10は、第四

の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

#### 【0081】

ステップS50において、画像処理装置40のWebサーバプログラム45がWebクライアント100からHTTPリクエストを受信すると、以降第三の実施の形態におけるステップS41からステップS43の手順と同様の手順をステップS51からS53において実行し、解凍プログラム44を呼び出す。

#### 【0082】

ステップS53に続いてステップS54に進み、解凍プログラム44は、解凍を要求された圧縮XSLファイルをキャッシュ管理テーブルから検索し、解凍対象の圧縮XSLファイルがキャッシュ管理テーブルに登録されているか否かを判断する。キャッシュ管理テーブルの詳細については後述するが、キャッシュ管理テーブルは、既に解凍されRAM42に格納されている圧縮XSLファイルの一覧を管理しているものである。解凍対象の圧縮XSLファイルがキャッシュ管理テーブルに登録されている場合、即ち既にRAM42に解凍されている場合はステップS58に進み、登録されていない場合、即ちまだ解凍されていない場合はステップS55に進む。

#### 【0083】

ステップS55及びステップS56において、解凍プログラム44は、第三の実施の形態におけるステップS44及びステップ45における手順と同様の手順で、指定された圧縮XSLファイルを解凍する。

#### 【0084】

ステップS56に続いてステップS57に進み、解凍プログラム44は、解凍した圧縮XSLファイルをキャッシュ管理テーブルに登録する。

#### 【0085】

ステップS58以降において、画像処理装置40は、第三の実施の形態におけるステップS47及びS48と同様の手順でWebページを生成し、Webクライアント100に送信する。

#### 【0086】

更に、ステップS54及びS59等によるキャッシュ管理テーブルの検索、更新処理の詳細について説明する。なお、キャッシュ管理テーブルは図11に示すようなリスト構造を利用して実装することができる。

#### 【0087】

図11は、リスト構造によるキャッシュ管理テーブルの実装例を示す図である。図11において、リスト50は、キャッシュエントリ501、502及び503を有している。各キャッシュエントリは、圧縮XSLファイルのファイル名である「ファイル名」と、その圧縮XSLファイルが解凍されたRAM42上のアドレスである「RAM上のアドレス」と、「次のエントリへのポインタ」とをデータ項目として有している。

#### 【0088】

以下、キャッシュ管理テーブルを図11のようなリスト構造で実装した場合のキャッシュ管理テーブルの検索、更新処理の詳細について説明する。図12は、キャッシュ管理テーブルの検索及びLRU法による更新処理を説明するためのフローチャート図である。なお、図12のフローチャートは、図10のフローチャートにおけるステップS54からS57までの処理に該当する。

#### 【0089】

ステップS100において、解凍プログラム44は、リスト50から最初のキャッシュエントリを取り出し、キャッシュエントリのファイル名が解凍対象の圧縮XSLファイルのファイル名と同じかどうか判断する(S101)。ファイル名が同じ場合はステップS102に進み、ファイル名が異なる場合はステップS103に進む。

#### 【0090】

ステップS103において、解凍プログラム44は、既に取り出したキャッシュエントリの「次のエントリへのポインタ」を参照することにより、リスト50から次のキャッシュエントリを取り出す。ステップS103に続いてステップS104に進み、解凍プログラム44は、キャッシュエントリがもう存在しないかどうかを判断する。この判断は、既に取り出したキャッシュエントリの「次のエントリポインタ」の値がNULLか否かによって行うことができ、解凍プログラ

ムはNULLだったら次のキャッシュエントリは存在しないと判断する。

#### 【0091】

キャッシュエントリが存在する場合は、ステップS101に進み、再び解凍対象のファイル名とキャッシュエントリのファイル名を比較する。ファイル名が同一の場合は、解凍対象の圧縮XSLファイルは既にRAM42に解凍されているため、ステップS102に進み、現在処理対象のキャッシュエントリをリストの先頭に繋ぎかえて処理を終了する。ここで、処理対象のキャッシュエントリをリストの先頭とするのは、次回の検索の際に使用されたXSLデータに対するキャッシュエントリを最初の比較対象とするためである。なお、処理を終了する際に解凍プログラム44は、解凍されたXSLデータのRAM42上におけるアドレスをキャッシュエントリから取り出して、呼び出し元のXSLTプロセッサ47に対して通知する。

#### 【0092】

ステップS104において、リスト50内に解凍対象とする圧縮XSLファイルに対応するキャッシュエントリを発見することができなかった場合は、ステップS105に進み、解凍プログラム44は解凍対象の圧縮XSLファイルをRAM41から解凍し、RAM42に格納する。この処理は、図10におけるステップS58及びステップS59に対応する。

#### 【0093】

ステップS105に続いてステップS106に進み、解凍プログラム44は、ステップS105においてRAM42に格納されたXSLデータに対応するキャッシュエントリを生成し、リスト50の先頭に追加する。

#### 【0094】

ステップS106に続いてステップS107に進み、解凍プログラム44は、リスト50のキャッシュエントリの数が所定の上限を超えているかどうかを判断する。この上限はRAM42の容量等に応じて適宜定めればよい。上限を超えていない場合はそのまま処理を終了し、超えている場合は、リスト50の末尾のキャッシュエントリを削除し、さらにその直前のエントリの「次のエントリへのポインタ」の値をNULLにして、処理を終了する。ここで、末尾のキャッシュエ

ントリを削除の対象とするのは、最も長い間利用されていないXSLデータをRAM42から削除するためである。なお、処理を終了する際に解凍プログラム44は、新たに解凍したXSLデータのRAM42上におけるアドレスを、呼び出し元のXSLTプロセッサに対して通知する。

#### 【0095】

図12においては、リスト50においてキャッシングエントリの数の上限を超えた場合は、最も長い間使用されていないキャッシングエントリが削除、即ち最も長い間使用されていないXSLデータをRAM42から削除するようにしているが（LRU（Least Recently Used）法）、リスト50内において最初に生成されたキャッシングエントリを削除、即ち解凍された時期が最も早いXSLデータをRAM42から削除するようにしてもよい（FIFO（First-In First-Out）法）。その場合のフローチャートを図13に示す。

#### 【0096】

図13は、キャッシング管理テーブルの検索及びFIFO法による更新処理を説明するためのフローチャート図である。図13において、解凍プログラム44は、解凍対象とする圧縮XSLファイルに対応するキャッシングエントリを見つける場合には（S102でYesの場合）、そのキャッシングエントリをリスト50の先頭にすることなく処理を終了する。また、新たに圧縮XSL解凍し（S204）、それに対するキャッシングエントリを生成した際には、そのキャッシングエントリはリストの末尾に追加する（S205）。更に、キャッシングエントリの数が上限を超えた場合には（S206でYesの場合）、先頭のキャッシングエントリを削除する（S207）。これにより、FIFO法によりリスト50を更新することができる。

#### 【0097】

上述したように、第四の実施の形態における画像処理装置40によれば、ROM41には圧縮されたXSLファイルを格納するため、ROM41の容量を節約することができる。また、画像処理装置40は、Webクライアント100からのHTTPリクエストを受信した際に、HTTPリクエストの応答としてのWebページの生成に必要な圧縮XSLファイルのみをRAM42に解凍するため、

RAM42の領域を節約することができる。更に、一度解凍した圧縮XSLファイルは、XSLデータとしてRAM42に格納されているため、以降にWebクライアント100からの要求に対して同じXSLデータを利用する場合は、レスポンスの悪化を防止することができる。

#### 【0098】

なお、上述したそれぞれの実施の形態においては、Webサーバとして画像処理装置を適用した例について説明したが、一般的な情報処理装置をWebサーバとして適用しても同様の効果を得ることができる。

#### 【0099】

以上、本発明の好ましい実施例について詳述したが、本発明は係る特定の実施形態に限定されるものではなく、特許請求の範囲に記載された本発明の要旨の範囲内において、種々の変形・変更が可能である。

#### 【0100】

##### 【発明の効果】

上述の如く、本発明によれば、Webページを生成するための書式情報を圧縮して記憶手段に格納しておくため、書式情報を使用したWebページの生成を少ない記憶容量で行うことができる。

##### 【図面の簡単な説明】

###### 【図1】

本発明の実施の形態における画像処理装置の機能構成例を示す図である。

###### 【図2】

本発明の第一の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

###### 【図3】

第一の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

###### 【図4】

本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

**【図5】**

第二の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

**【図6】**

本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

**【図7】**

第三の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

**【図8】**

圧縮XSLファイル対応テーブルの一例を示す図である。

**【図9】**

本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図である。

**【図10】**

第四の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図である。

**【図11】**

リスト構造によるキャッシュ管理テーブルの実装例を示す図である。

**【図12】**

キャッシュ管理テーブルの検索及びLRU法による更新処理を説明するためのフローチャート図である。

**【図13】**

キャッシュ管理テーブルの検索及びFIFO法による更新処理を説明するためのフローチャート図である。

**【符号の説明】**

10、20、30、40 画像処理装置

11、21、31、41 ROM

12、22、32、42 RAM

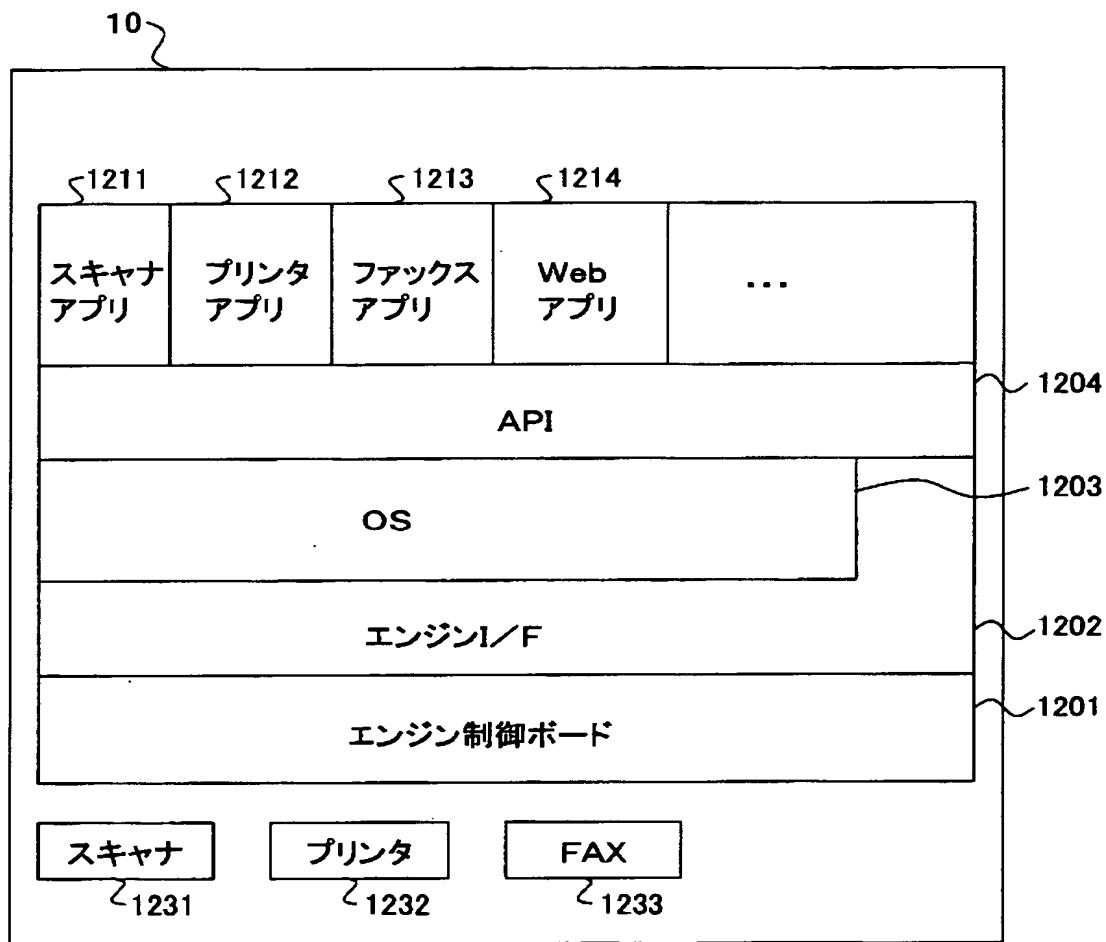
13、23、33、43 初期化プログラム  
14、24、34、44 解凍プログラム  
15、25、35、45 Web サーバプログラム  
16、26、36、46 Web アプリケーション  
17、27、37、47 XSLT プロセッサ  
50 リスト  
100 Web クライアント  
361 圧縮XSL ファイル対応テーブル  
501、502、503 キャッシュエントリ  
1201 エンジン制御ボード 1202 エンジン I/F  
1203 OS 1204 API  
1211 プリンタアプリ 1212 スキャナアプリ  
1213 ファックスアプリ 1214 Web アプリ  
1231 スキャナ 1232 プリンタ  
1233 FAX

【書類名】

図面

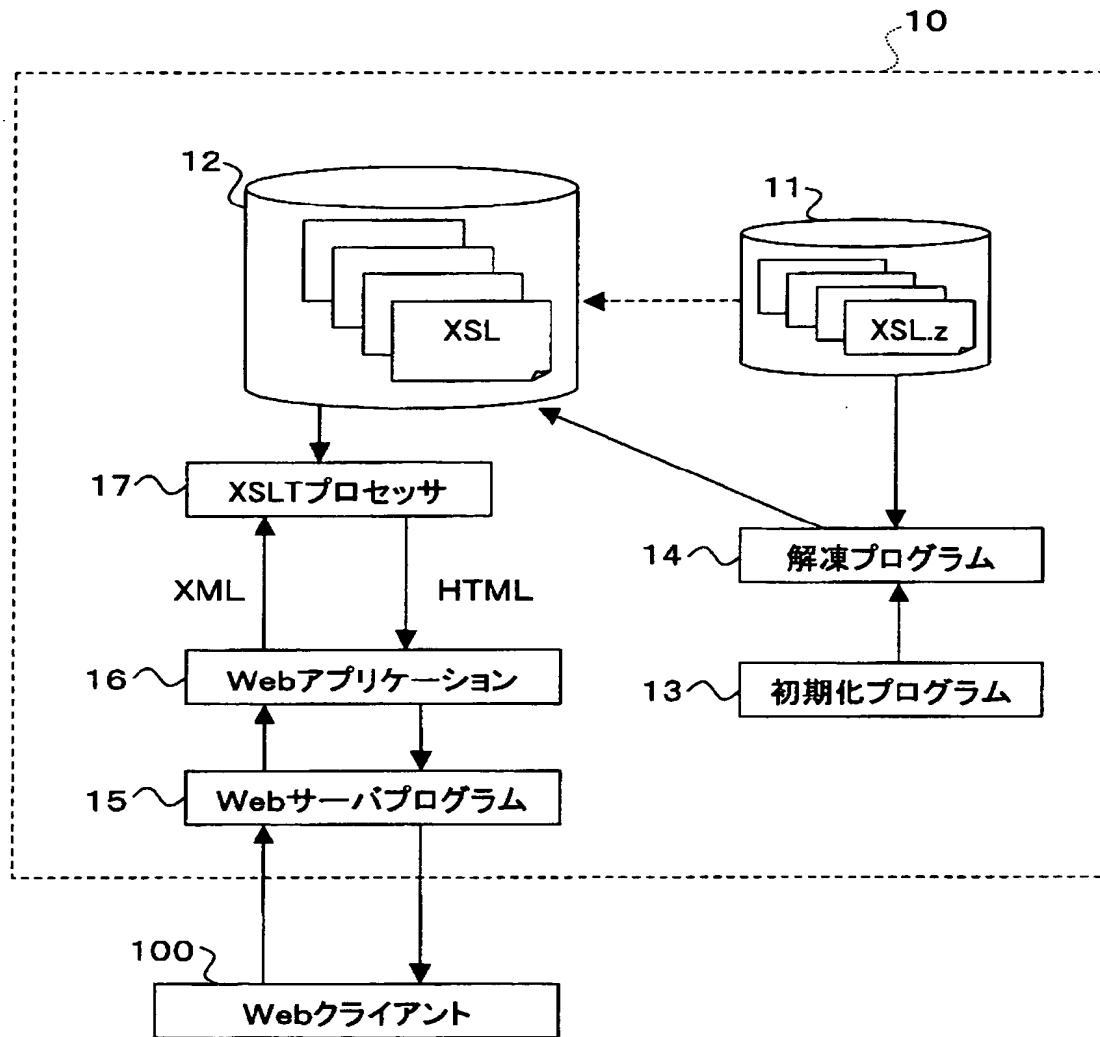
【図 1】

## 本発明の実施の形態における画像処理装置の機能構成例を示す図



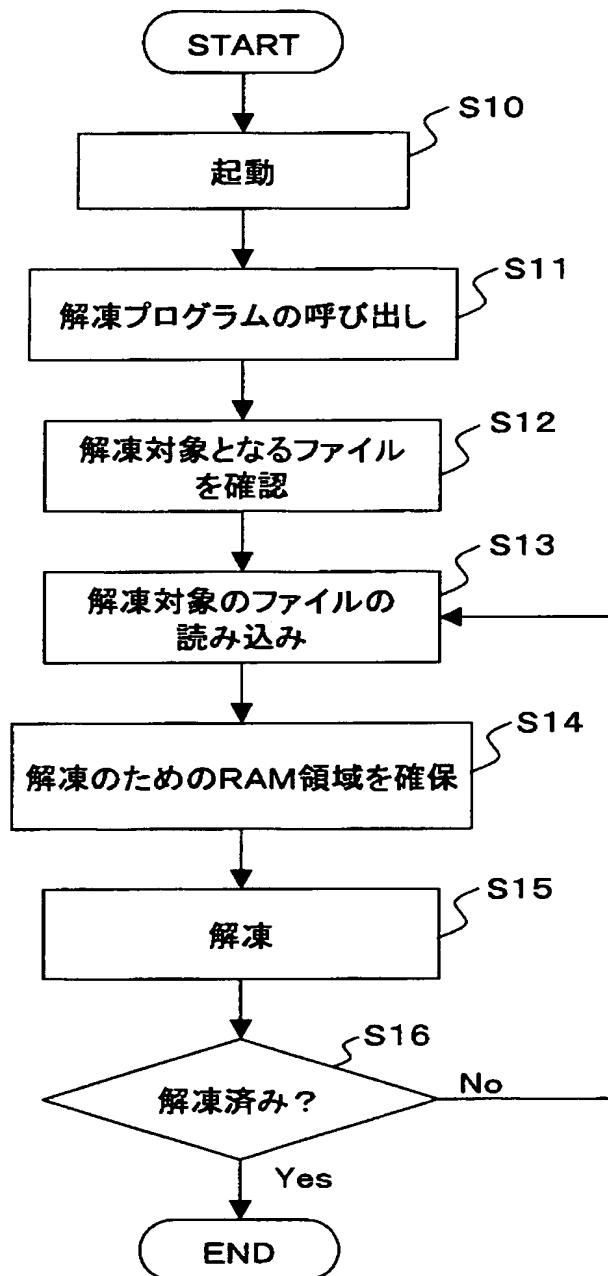
【図2】

## 本発明の第一の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



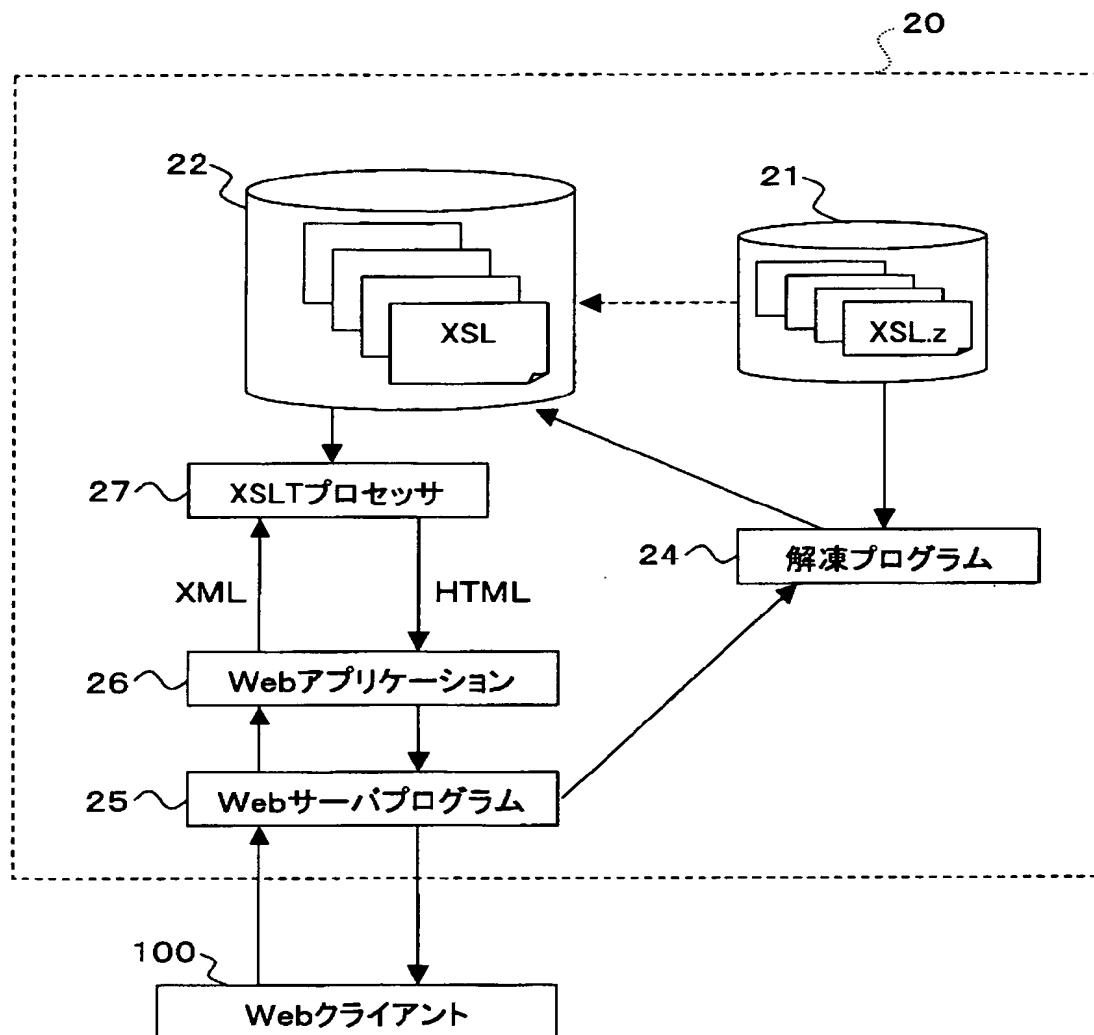
【図3】

第一の実施の形態における画像処理装置のXSLファイルの解凍処理を説明するためのフローチャート図



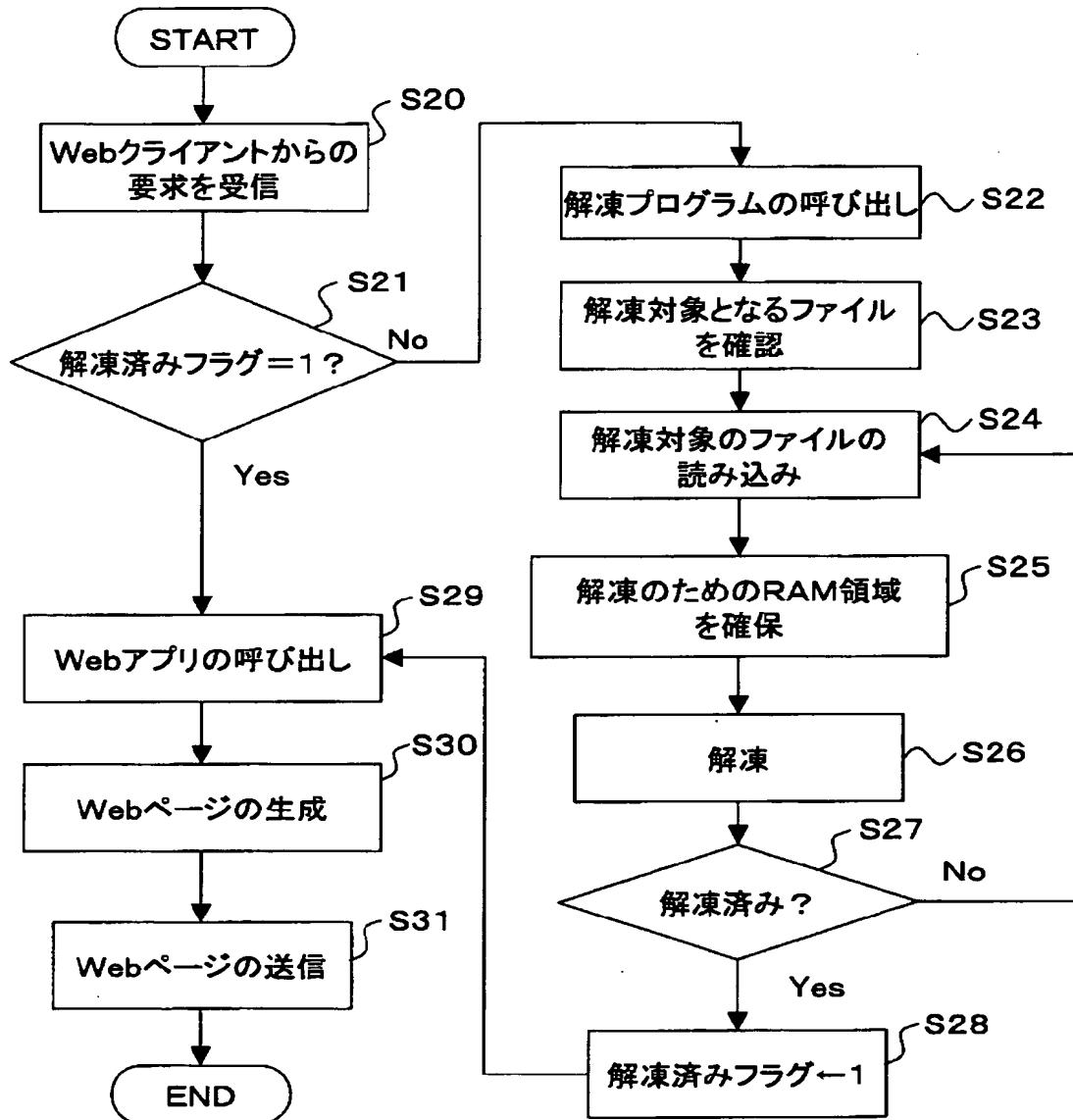
【図4】

## 本発明の第二の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



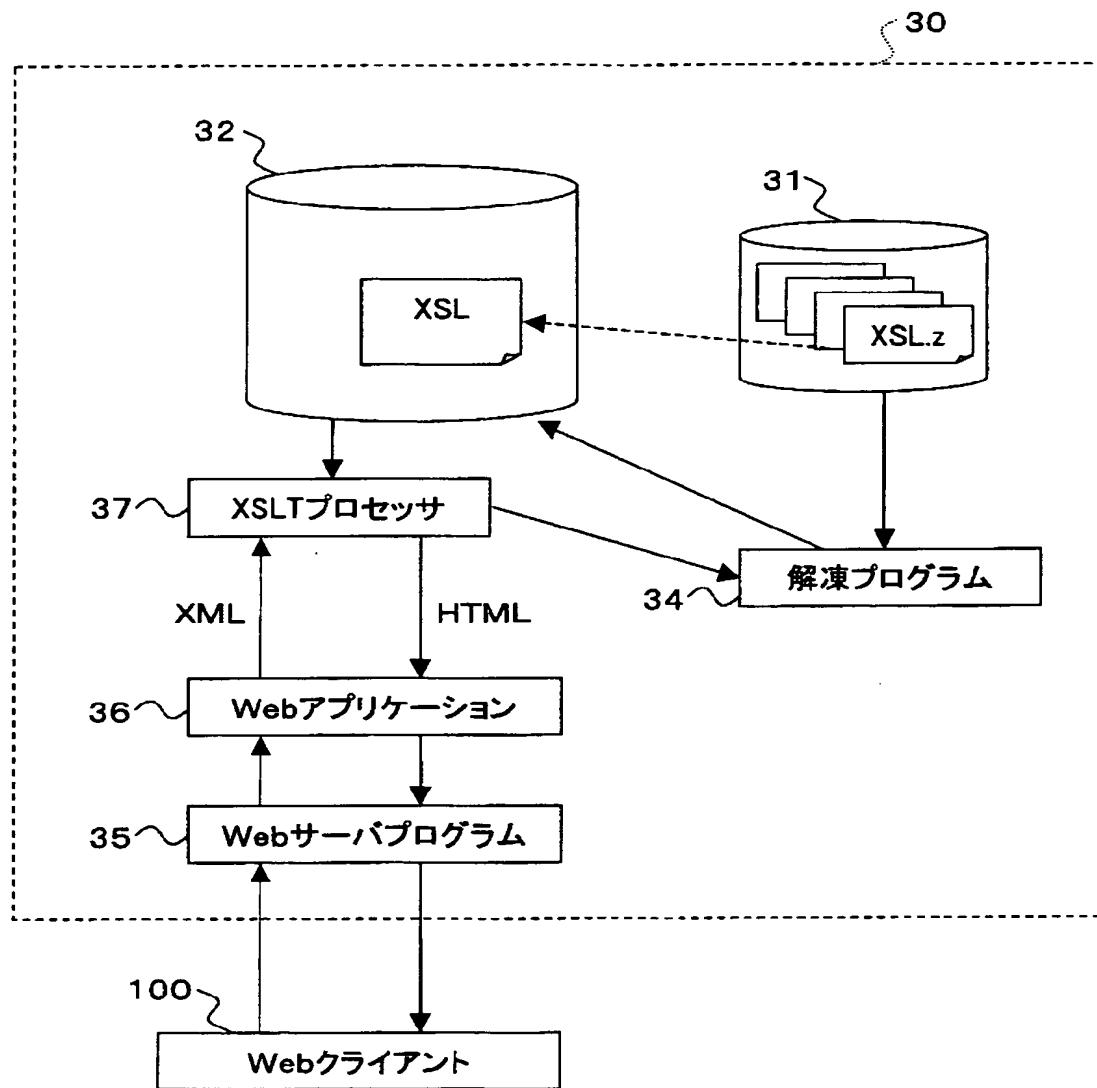
【図5】

第二の実施の形態における画像処理装置のXSLファイルの  
解凍処理を説明するためのフローチャート図



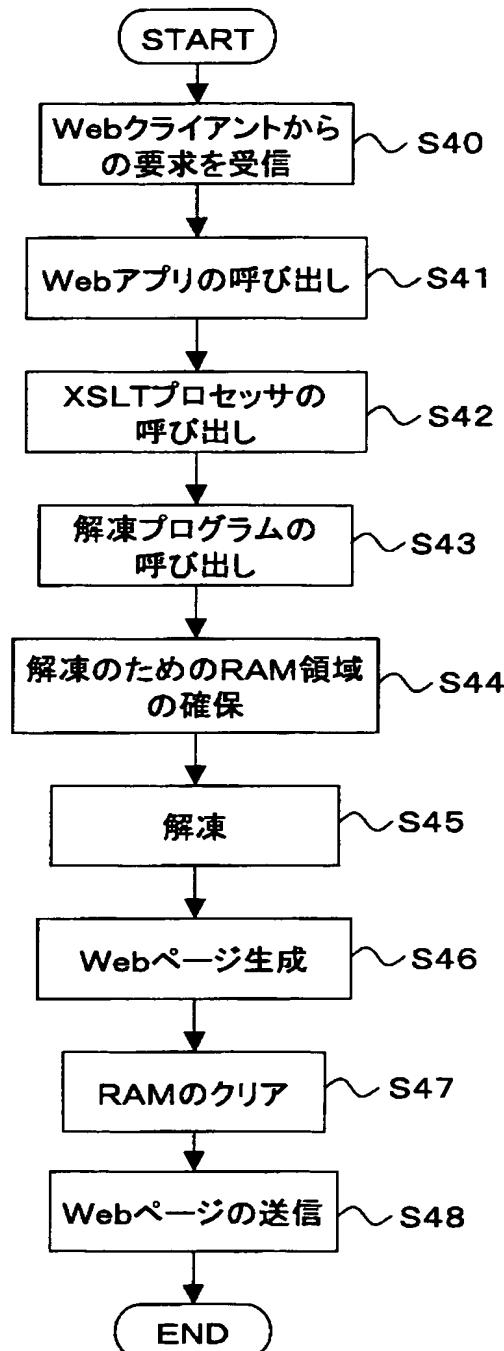
【図6】

本発明の第三の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



【図7】

第三の実施の形態における画像処理装置のXSLファイルの  
解凍処理を説明するためのフローチャート図



【図8】

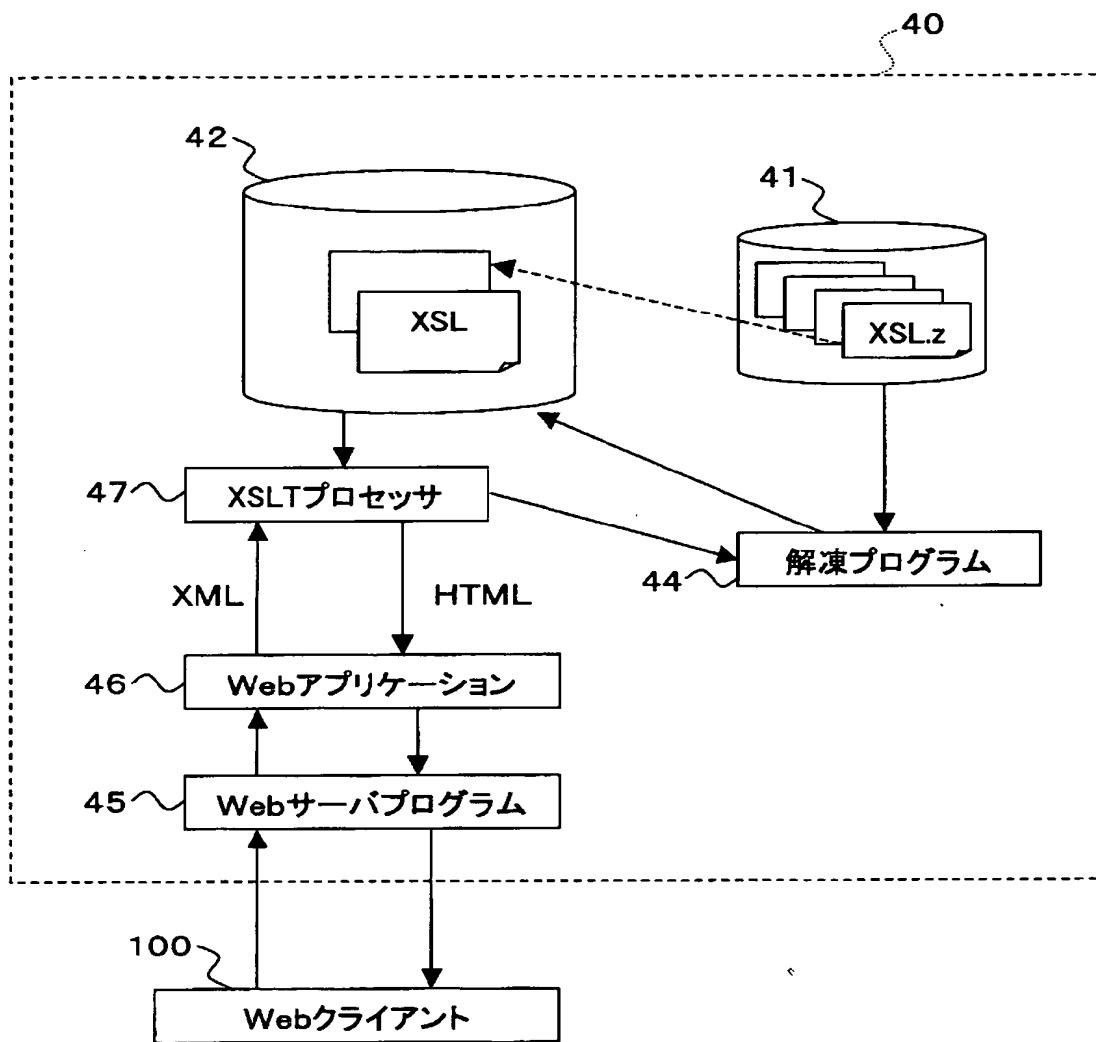
圧縮XSLファイル対応テーブルの一例を示す図

361

URL	XSL
topPage. cgi	topPage. xsl. z
getStatus. cgi	getStatus. xsl. z
inTray. cgi	inTray. xsl. z
outTray. cgi	outTray. xsl. z

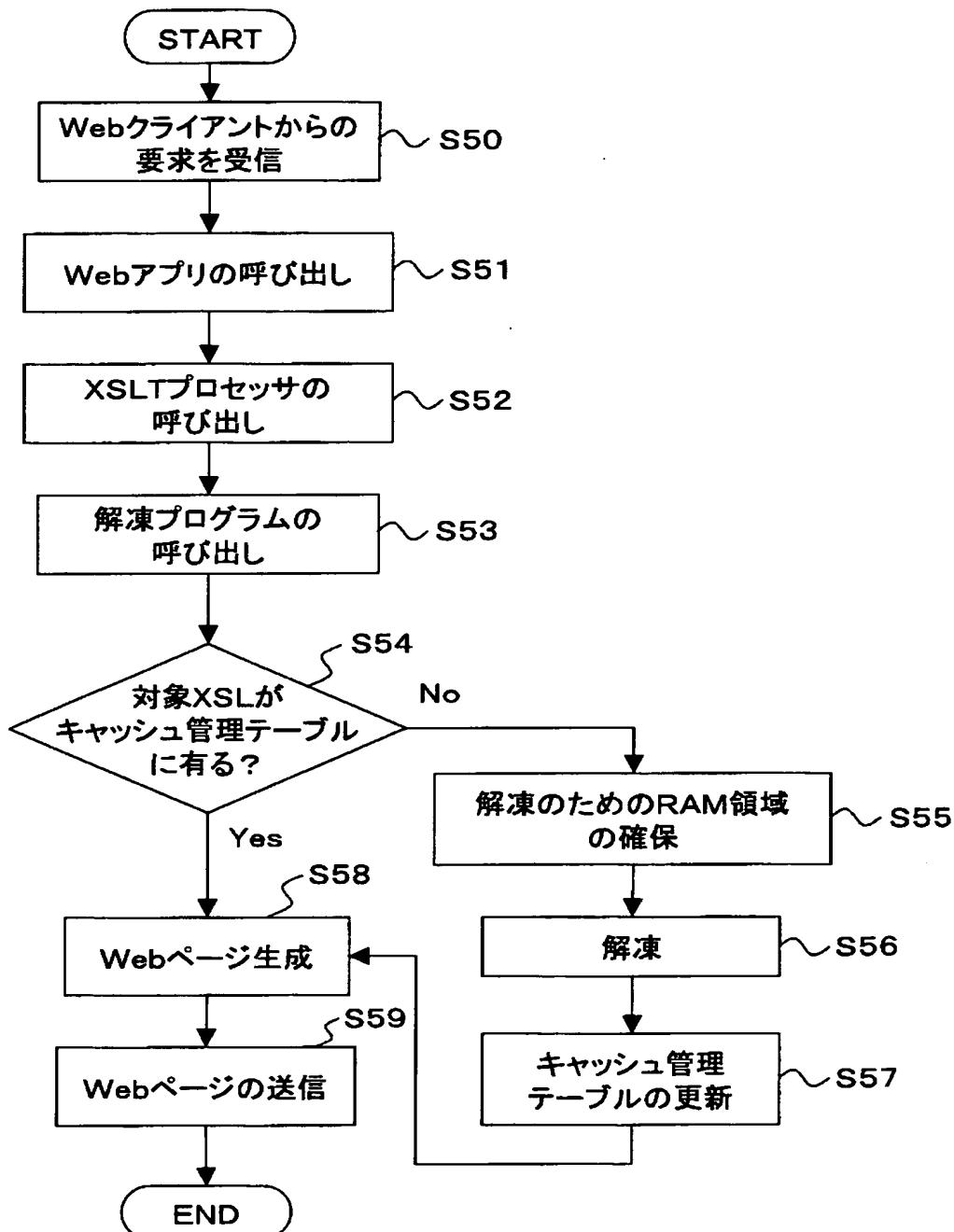
【図9】

## 本発明の第四の実施の形態におけるWebサーバとしての画像処理装置の機能構成例を示す図



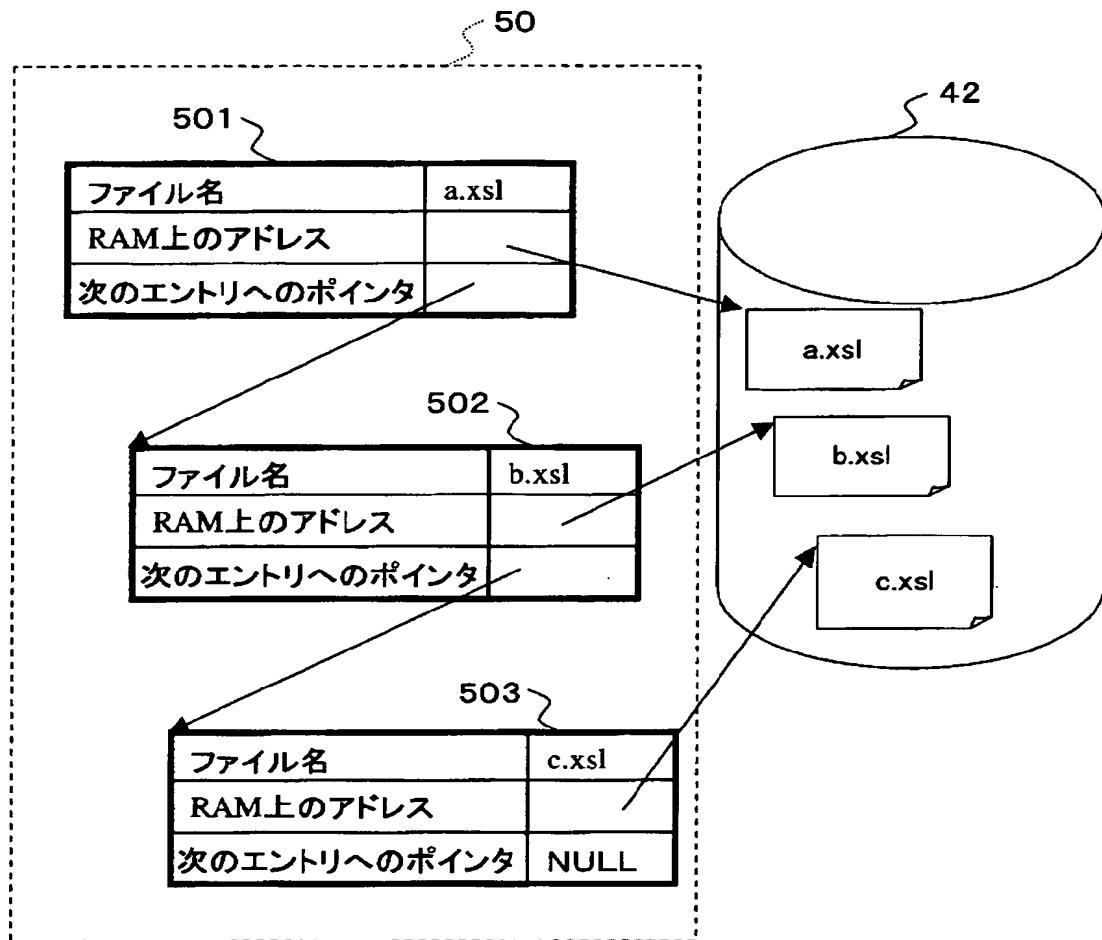
【図10】

第四の実施の形態における画像処理装置のXSLファイルの  
解凍処理を説明するためのフローチャート図



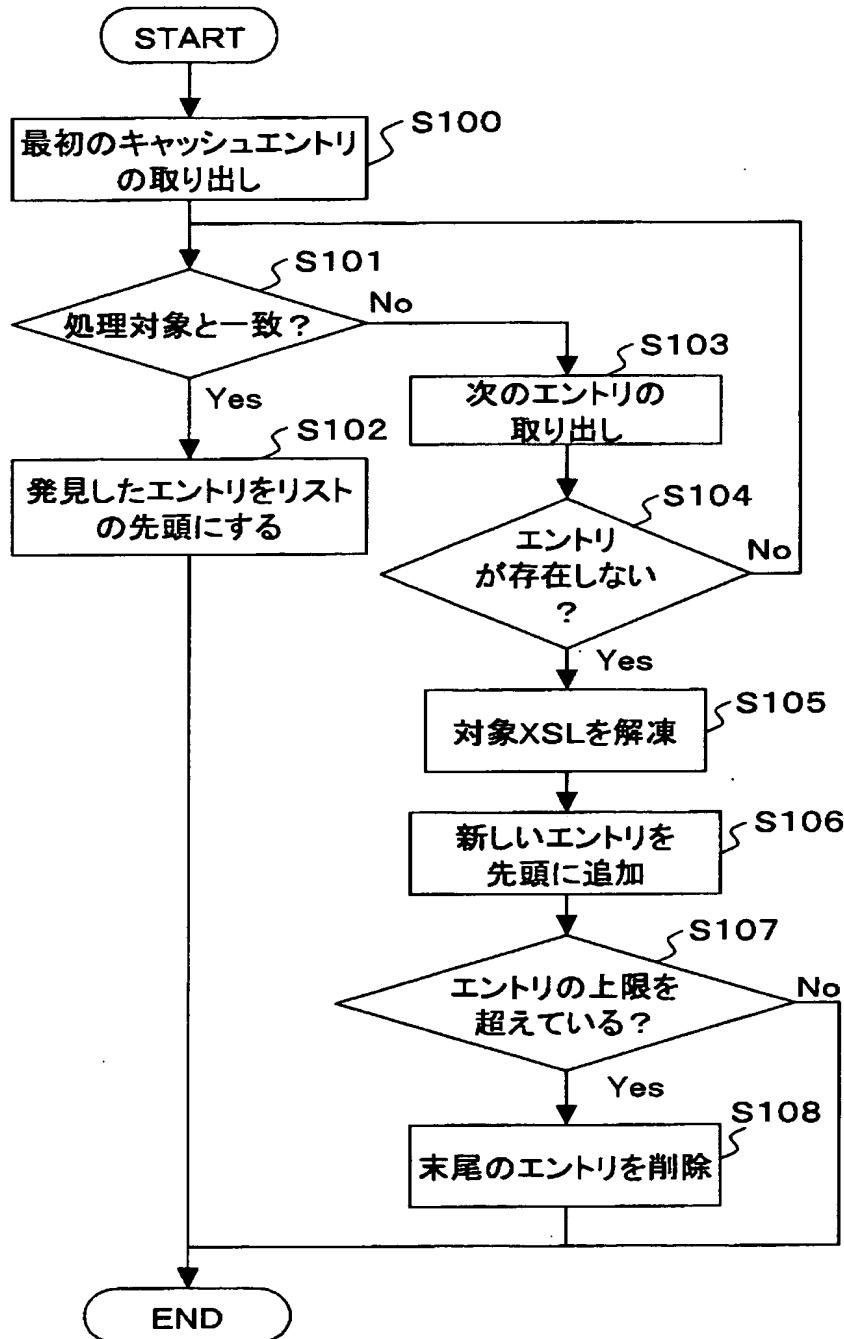
【図11】

リスト構造によるキャッシュテーブルの実装例を示す図



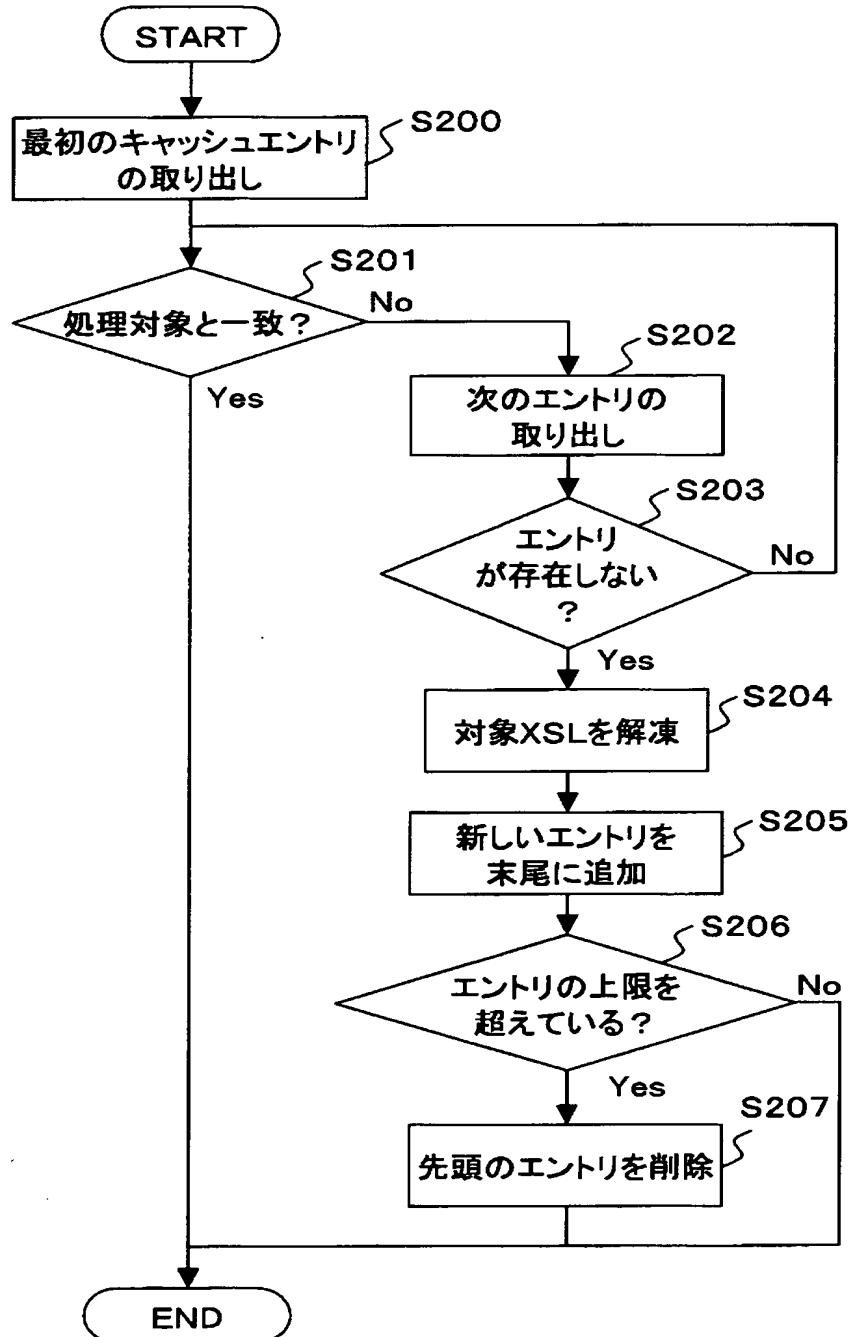
【図12】

キャッシュ管理テーブルの検索及びLRU法による  
更新処理を説明するためのフローチャート図



【図13】

キャッシュ管理テーブルの検索及びFIFO法による更新処理を実装する場合の説明するためのフローチャート図



【書類名】 要約書

【要約】

【課題】 書式情報を使用したWebページの生成を少ない記憶容量で行うことができる画像処理装置の提供を目的とする。

【解決手段】 ネットワークを介して接続される端末からの要求を受信し、前記端末に表示される情報を表すWebページを前記端末に送信する送受信制御手段と、圧縮されている複数の書式情報ファイルを格納する第一の記憶手段と、該第一の記憶手段が格納する前記圧縮されている複数の書式情報ファイルを書式情報データに解凍する解凍手段と、前記書式情報データを格納する第二の記憶手段と、前記書式情報データを使用して前記Webページを生成するWebページ生成手段とを有し、該Webページ生成手段により生成された前記Webページを前記送受信制御手段により前記端末に送信することにより上記課題を解決する。

【選択図】 図2

特願2002-258190

出願人履歴情報

識別番号 [000006747]

1. 変更年月日 1990年 8月24日  
[変更理由] 新規登録  
住 所 東京都大田区中馬込1丁目3番6号  
氏 名 株式会社リコー

2. 変更年月日 2002年 5月17日  
[変更理由] 住所変更  
住 所 東京都大田区中馬込1丁目3番6号  
氏 名 株式会社リコー